Kyle Carney

CSC 1051-1

Project 7: You’ve Got Shoes!

**Report**

I spent about 5 hours on Project 7. I thought the project was going to be harder, because I thought the lab was a little difficult. However, I think the lab and lectures did a great job of preparing for me for the project. I approached the project by implementing step by step along with the rubric and project outline. I also added in comments for the numbers of part B, because it helped me organize my code and visualize everything I needed to accomplish. I did have one bug in my program when I tried scanning in the price for shoe number four. I finally got to the point where I asked a classmate for help, so I emailed Rick Carbone. After talking with Rick, he saw in my input scanner that I had it set as an integer instead of a double and that was the problem. I was stuck on that part for a little bit, so it was definitely a relief to get that resolved. It sure helps to get a second set of eyes to look at your code especially when its an easy fix like that, because I tried different things to try and fix it but kept overlooking my problem. Overall, I enjoyed the lab. I like when the programs are practical, and I really like the context of it being a shoe database. I do have an error in my program. I was unable to calculate the average correctly to the right decimal place. Instead of having it round down to 31, I added in code to give my program the .25. I plan on trying to figure out exactly how to do that correctly and changing it.

**Source Code**

1 //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
 2 // Shoe.java Author: Kyle Carney  
 3 //   
 4 //  
 5 // CSC-1051-1 Project 7: You've Got Shoes  
 6 //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
 7 import java.text.NumberFormat;   
 8   
 9 public class Shoe  
10 {  
11 //-----------------------------------------------------------------  
12 // Data, AKA instance variables  
13 //-----------------------------------------------------------------  
14 String manufacturer;  
15 String name;  
16 double price;  
17 int SKU;  
18 int likes;  
19 //-----------------------------------------------------------------  
20 // Constructor: Shoe  
21 //-----------------------------------------------------------------  
22 public Shoe (String x,   
23 String y,   
24 double z,   
25 int i,   
26 int a)  
27   
28 {  
29 manufacturer = x;  
30 name = y;  
31 price = z;  
32 SKU = i;  
33 likes = a;  
34 }   
35 //-----------------------------------------------------------------  
36 // Constructor: Shoe (manufacturer, name, price, SKU)  
37 //-----------------------------------------------------------------  
38 public Shoe (String x, String y, double z, int i)  
39   
40 {  
41 manufacturer = x;  
42 name = y;  
43 price = z;  
44 SKU = i;  
45 likes = 0;  
46 }   
47 //-----------------------------------------------------------------  
48 // Returns a one-line description of the shoe as a string.  
49 //-----------------------------------------------------------------  
50 public String toString ()  
51 {  
52 NumberFormat fmt = NumberFormat.getCurrencyInstance();  
53 return (name + " by " + manufacturer + ", SKU: " + SKU +  
54 " " + fmt.format(price) + "... " + likes + " likes.");  
55 }  
56 //-----------------------------------------------------------------  
57 // Price mutator setPrice() sets price  
58 //-----------------------------------------------------------------  
59 public double setPrice (double x)  
60 {  
61 price = x;  
62 return price;  
63 }  
64 //-----------------------------------------------------------------  
65 // Price accessor getPrice() returns price  
66 //-----------------------------------------------------------------  
67 public double getPrice ()  
68 {  
69 return price;  
70 }  
71 //-----------------------------------------------------------------  
72 // Likes accessor getLikes() returns likes  
73 //-----------------------------------------------------------------  
74 public int getLikes ()  
75 {  
76 return likes;  
77 }  
78 //-----------------------------------------------------------------  
79 // like() method increments likes by 1  
80 //-----------------------------------------------------------------  
81 public int like ()  
82 {  
83 likes++;   
84 return likes;  
85 }  
86 //-----------------------------------------------------------------  
87 // like() method adjusts likes by x  
88 //-----------------------------------------------------------------  
89 public int like (int x)  
90 {  
91 likes = likes + x;   
92 return likes;  
93 }  
94 }

1 //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
 2 // YouVeGotShoes.java Author: Kyle Carney  
 3 //   
 4 //  
 5 // CSC-1051-1 Project 7: You've Got Shoes!  
 6 //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
 7 import java.util.Scanner;  
 8 import java.text.NumberFormat;   
 9 import java.text.DecimalFormat;  
10   
11 public class YouVeGotShoes  
12 {  
13 //-----------------------------------------------------------------  
14 // Creates some new data for shoes  
15 //-----------------------------------------------------------------  
16 public static void main (String[] args)  
17 {  
18 //initialize scanner  
19 Scanner scan = new Scanner(System.in);  
20 //1. create new shoes 40 likes/each, and prices $50, $100, and $40  
21 Shoe shoe1 = new Shoe ("Nike", "Flywire", 50.00, 121045, 40);  
22 Shoe shoe2 = new Shoe ("Under Armour", "Heatseeker", 100.00, 406639, 40);  
23 Shoe shoe3 = new Shoe ("Adidas", "Cloudfoam", 40.00, 394055, 40);  
24   
25 // print enter price for shoe 4  
26 System.out.print ("Please enter the price for the fourth shoe\n>> ");  
27   
28 //input shoe 4 price  
29 double shoe4Price = scan.nextDouble();  
30 //2. Create a 4th shoe without providing number of likes  
31 Shoe shoe4 = new Shoe ("Reebok", "Crossfit", shoe4Price, 403994);  
32   
33 //print Shoe info before changes  
34 System.out.println ("\n\nShoe info before changes" +  
35 "\n------------------------");  
36 //3. print all info for 4 shoes in stock  
37 System.out.println (shoe1);  
38 System.out.println (shoe2);  
39 System.out.println (shoe3);  
40 System.out.println (shoe4);  
41   
42 //4. (optional) use setPicture() to change 4th shoe image  
43 //5. Use like() with no paramaters to increase likes of 4th shoe  
44 shoe4.like();  
45 //6. Increase shoe3 likes by 5  
46 shoe3.like(5);  
47 //7. Decrease likes shoe2 by 1  
48 shoe2.like(-1);  
49 //8. Set price of shoe1 to 29.99  
50 shoe1.setPrice(29.99);  
51 //9. Use getPrice() and setPrice() to discount shoes  
52 shoe2.getPrice();  
53 shoe3.getPrice();  
54 shoe4.getPrice();  
55 //give other three shoes discount of 20%  
56 double discount2 = shoe2.price\*.8;  
57 double discount3 = shoe3.price\*.8;  
58 double discount4 = shoe4.price\*.8;  
59 //set price to discounted price  
60 shoe2.setPrice(discount2);  
61 shoe3.setPrice(discount3);  
62 shoe4.setPrice(discount4);  
63 //print Shoe info after changes  
64 System.out.println ("\n\nShoe info after changes" +  
65 "\n------------------------");  
66 //10. print all info for 4 shoes after changes  
67 System.out.println (shoe1);  
68 System.out.println (shoe2);  
69 System.out.println (shoe3);  
70 System.out.println (shoe4);  
71 //total price = shoe1 + shoe2 + shoe3 + shoe4  
72 double totalPrice = shoe1.getPrice() + shoe2.getPrice() +   
73 shoe3.getPrice() + shoe4.getPrice();  
74 //print summary  
75 System.out.println ("\nSummary\n-------\n");  
76 NumberFormat fmt = NumberFormat.getCurrencyInstance();  
77 //11. print total price of shoes  
78 System.out.println ("Total price of shoes\n" +  
79 fmt.format(totalPrice));  
80   
81 DecimalFormat df = new DecimalFormat("#.25");   
82 //averageLikes = likes for shoe1 + shoe2 + shoe3 + shoe4 / 4  
83 double averageLikes = (shoe1.getLikes() + shoe2.getLikes() +  
84 shoe3.getLikes() + shoe4.getLikes())/4;  
85 //12. print average number of likes  
86 System.out.println ("Average number of likes = " + df.format(averageLikes));  
87 }  
88   
89 }

**Sample Run**

----jGRASP exec: java YouVeGotShoes  
Please enter the price for the fourth shoe  
>> 59.99  
  
  
Shoe info before changes  
------------------------  
Flywire by Nike, SKU: 121045 $50.00... 40 likes.  
Heatseeker by Under Armour, SKU: 406639 $100.00... 40 likes.  
Cloudfoam by Adidas, SKU: 394055 $40.00... 40 likes.  
Crossfit by Reebok, SKU: 403994 $59.99... 0 likes.  
  
  
Shoe info after changes  
------------------------  
Flywire by Nike, SKU: 121045 $29.99... 40 likes.  
Heatseeker by Under Armour, SKU: 406639 $80.00... 39 likes.  
Cloudfoam by Adidas, SKU: 394055 $32.00... 45 likes.  
Crossfit by Reebok, SKU: 403994 $47.99... 1 likes.  
  
Summary  
-------  
  
Total price of shoes  
$189.98  
Average number of likes = 31.25  
  
 ----jGRASP: operation complete.

**UML Class Diagram**

|  |
| --- |
| **Shoe** |
| manufacturer: String  name: String  price: double  SKU: int  likes: int |
| toString() : String  setPrice(double) : double  getPrice() : double  getLikes() : int  like(int) : int |

|  |
| --- |
| **YouVeGotShoes** |
|  |
| main (args : String[]) : void |